Insights

Is Too Much Safety Safe? The Danger of Large Safety Arguments

There is a concerning trend developing in industry where a system is often deemed 'safe' if it is accompanied by a large amount of paperwork. However, in many cases these arguments fail to provide verifiable evidence that the system being delivered is actually safe. So, how do you avoid the assurance pitfalls and deliver a meaningful safety case?

Every day, millions of people catch a train. For the most part, everyone takes for granted that the journey will be safe. But how are we assuring that journey will be safe?

Under rail safety national law everyone involved in the design, commissioning, manufacture, supply, or installation of rail infrastructure has a duty to demonstrate that the safety risk of the delivered product has been eliminated or, where not possible to eliminate, reduced so far as is reasonably practicable (SFAIRP) [1]. We do this by providing a safety assurance argument or safety case for the system.

However, in many cases a system is deemed safe for use because it is accompanied by a mountain of expensive, time-consuming, and inconclusive paperwork; rather than a clear concise argument that explains why the system is safe.

This increasing focus across the industry on preparing paperwork to get through a "gate" is leading to safety assurance arguments that fail to adequately argue a system is safe.

This paper explores how we can deliver a safety argument for a system that is clear, concise, and cost-effective. An argument that is tailored for the system in question and, when read, leaves the reader understanding "why" the system is safe and convinced the risk is reduced SFAIRP.

The Problem

The author has observed that a number of the safety arguments being delivered today fail to provide a convincing argument the system being delivered is safe. The following sections capture a number of examples from the author's experience which illustrate some of the contributing factors to this trend.

Incomplete Safety Arguments

A contributing factor the author has often observed is safety arguments or safety cases which are missing a key element or simply present a very weak argument for that element.

Industry best practice as embodied in standards such as EN50129 [2] set out that a safety argument should at least include a Quality Management Report (QMR), a Safety Management Report (SMR), and a Technical Safety Argument (TSA) as depicted in Figure 1.

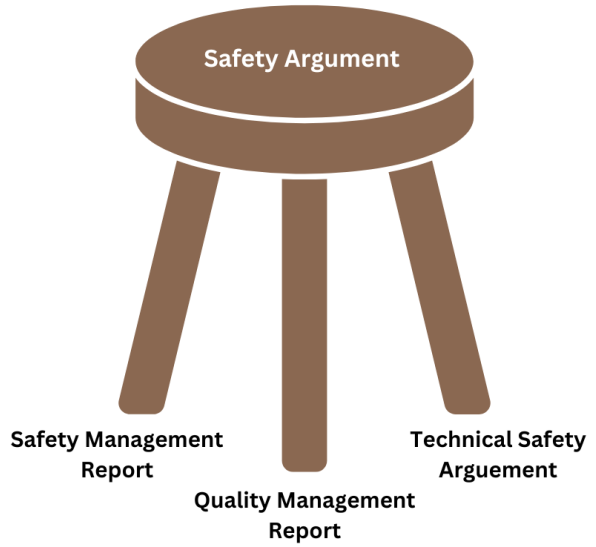


Fig. 1: A well-structured safety argument will withstand scrutiny

However, the author often finds that safety arguments focus on one or two of these elements, omitting the third. Typically, it is the technical safety argument that is lacking or non-existent due to the difficulty in clearly articulating what that argument is.

The Quality and Safety Management sections of the argument are typically of the form:

- We have planned to do activities A, B, and C
- We have completed A, B, and C and here is the resulting evidence of those activities.

This should generally be an easy and straight forward process to formulate, but even these arguments can be weak. The technical safety argument, on the other hand, attempts to explain why the delivered system, sub-system, or equipment item will safely perform its desired function.

Even with input from subject matter experts and a clear understanding of the system, this can be a challenging task and, as such, the author often finds safety arguments that, instead of being robust (see Figure 1), are weak and unconvincing upon close examination, which results in a collapse of the whole argument as illustrated in Figure 2.

It is important that the system safety engineers understand the product being delivered to ensure the technical safety arguments they write hold up under scrutiny.

A good safety argument needs to be well proportioned. It also needs to be complete. The author often sees well-formed arguments for the scope delivered by individual contract packages. However, the argument for how the system works across multiple packages is incomplete as it is missing the safety argument for that one little system in between.

Because that “one little system” is no one’s scope, or is someone else’s problem, the overarching argument is incomplete, and the system cannot be demonstrated as safe. A systems perspective on the safety of the full end product is needed.

Another example is when an argument is made that system A is safe in context X but then system A is used in context Y with an argument that it is still safe based on the evidence provided for context X.

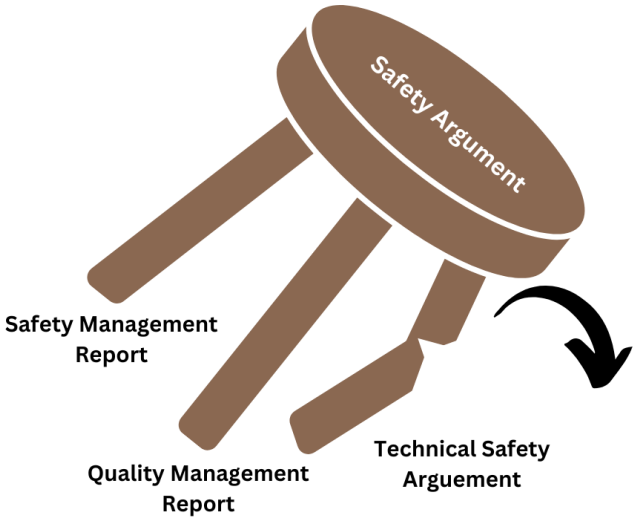


Fig. 2: A safety argument without strong legs will fail

"A good safety argument needs to be structured and to the point."

While this can be a valid risk acceptance principle (reference systems – see EN50126 [5] clause 6.3) care needs to be taken in its application and a clear argument needs to be made that the differences between context Y and context X are well understood and do not impact the safety argument.

Bloated Safety Arguments

A second contributing factor that the author observes is an unstructured and unnecessarily long-winded safety argument, full of unnecessary detail which hides the pertinent facts and leaves the reader, after having waded through hundreds of pages, asking the question “so what?”.

While a good safety argument may be hundreds or even thousands of pages, just because it took two reams of paper to print, it does not necessarily mean that the system is safe to use.

These bloated safety arguments can be the result of the argument being unstructured and lacking focus. However, even in a structured and focused argument, elements can become bloated when, for example, too much focus is given to a particular section, often to the detriment of the other sections (see above). The bloat can also result in conflicting information across multiple documents resulting in an incoherent safety argument. A good safety argument needs to be structured and to the point.

This can also occur when it is unclear why the argument is needed in the first place. Or alternatively why seventeen different hazard analyses are required to analyse a Business As Usual (BAU) activity.

The author has seen this occur when a contract defines the different types of safety analyses to be performed or the number of safety cases that are required based on what was done last time or to ensure the system is delivered “safely”.

However, this can often lead to a lot of potentially unnecessary work being completed. A project is better served by having a competent system safety engineer (see below) review the scope of work at the pre-contract stage and define what activities are required in the project system safety assurance plan, while demonstrating compliance with industry best practice (e.g. EN50126 [5], iESM [6]).

It is essential to plan early. Part of that plan needs to clearly define safety obligations, objectives, and targets. It also needs to establish a definitive system boundary from the beginning of the project.

This bloat can also occur when re-using an analysis from one similar system to another. For example, re-using a system safety assurance plan from the previous project for a new job without reviewing the scope/context or identifying the differences in design.

The team implementing the new job may find themselves performing unnecessary analysis or, even worse, failing to perform critical analyses required for the current project but not necessary (and hence not included) for the previous one which results in an incomplete argument (see above).

While it is good practice to review and take lessons learned from past projects, direct copy and paste should be avoided. There is no substitute for competent thinking.

Unsubstantiated Claims

A third contributing factor is to simply make a claim and even formulate an argument but fail to substantiate that claim and argument with evidence. Without available and auditable evidence, the safety argument may be of the finest logical and documentary structure but completely fail under scrutiny as depicted in Figure 3.

Fig. 3: A safety argument without evidence will not withstand scrutiny

A common example of this is simply restating what you have said in the plan as being complete without providing any evidence of that completion. Providing evidence need not be hard and can be as simple as providing cross references to where the evidence can be found.

A comparison of a statement in a plan with a bad and good example of evidence is illustrated in Figure 4.

Another example the author has recently observed is an over reliance on an independent safety assessment certificate which made a claim about a system that the project was relying on, but when challenged and investigated (requiring substantial effort) was found to be false thus raising questions about the validity of the whole assessment.

Therefore, it is important when making claims to make sure there is clear and substantive evidence available, even when relying on evidence such as an independent safety certificate.

This is not to say that relying on independent safety assessments is bad, only that if your argument relies on a certificate as evidence then be confident in its pedigree and investigate if anything appears out of place. There is no substitute for doing your own homework.

A good safety argument therefore needs to have accessible and auditable evidence to substantiate each claim.

Another contributor to unsubstantiated claims is moving or unclear goal posts. When formulating an argument, it is important to clearly communicate in your safety plan what claims you will make and what evidence in support of those claims will be provided in your safety case. When those goal posts move (i.e. the client determines just before commissioning a new claim needs to be made) the evidence that has been collected will

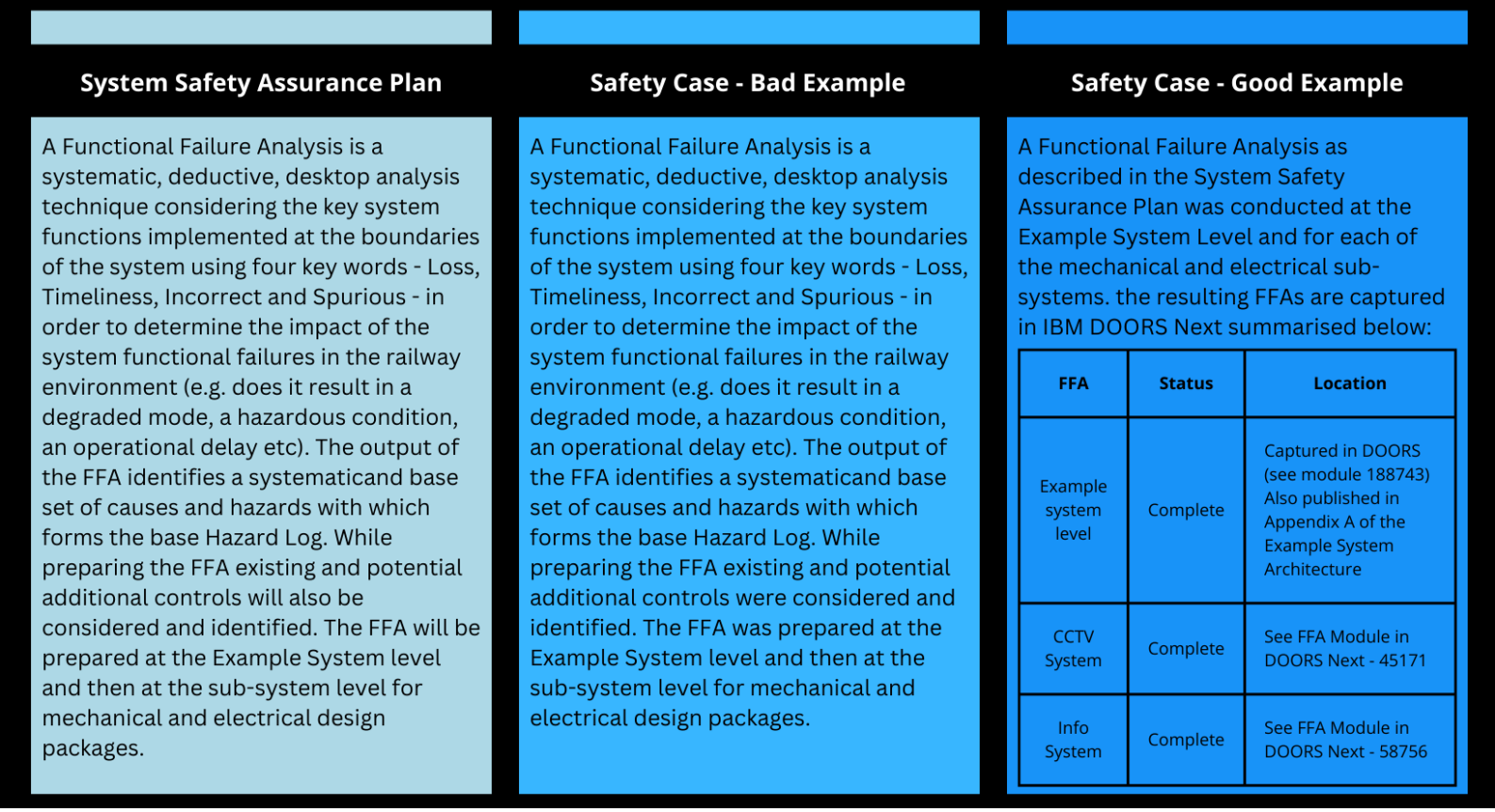


Fig. 4: Example of a claim without evidence compared to a claim with evidence

likely not be sufficient to substantiate the new claim. The author has seen this recently on a project, which caused substantial delays and headaches to all involved.

Linking to Achieve A Statistic

An important part of safety arguments today is being able provide evidence that all possible controls have been identified for each hazard, and that the identified controls have either been verified and validated prior to putting the system into service or a clear rationale is provided for why the control has been rejected.

To do this we use relational databases such as IBM DOORS Next (or even a good old spreadsheet for smaller projects). However, a fourth contributing factor that the author often sees is the application of links to meet the quality objectives that all controls mitigate at least one hazard and are satisfied by at least one requirement.

This can lead to situations where, to achieve a quality metric and get through a stage gate, the safety engineers spend an inordinate amount of time trying to find existing requirements that "satisfy" the identified controls rather than spending the limited time verifying that all hazards are captured and that the controls are in the design.

While there may be existing requirements (naturally, we want to avoid duplicating requirements) the activity often leads to vague and hard to follow linkages, which can result in difficulty following the evidence trail. If this happens, it can result in a lack of evidence that the control is in place. Such an example is shown in Figure 5.

While the link shown in Figure 5 can potentially be valid (i.e. the design does in fact incorporate the control) it would be much cleaner and potentially less time consuming both for the safety engineer and reviewers to simply derive requirements that are then evidenced in the design that the more specific control is adopted.

A challenge here is that a control is often captured in risk workshops as the design decision and a good requirement should not include implementation in it, so a balance does need to be made here. However, the point remains that, while traceability from hazard to cause to control

requirement to design artefact to V&V evidence is essential, it in itself does not make the system safe.

Care needs to be taken when setting up hazard and requirement databases to provide the needed evidence trail while avoiding an inordinate amount of time instantiating links that add no value to the safety of the implemented system. A hazard log filled with bad links reduces the reader's confidence in the safety case and the overarching veracity of the entire argument.

A good safety argument is supported by a well-structured hazard log which has been planned well in advance. It is important that safety controls/ requirements generated are clear and can be correctly interpreted and shown to be appropriately implemented.

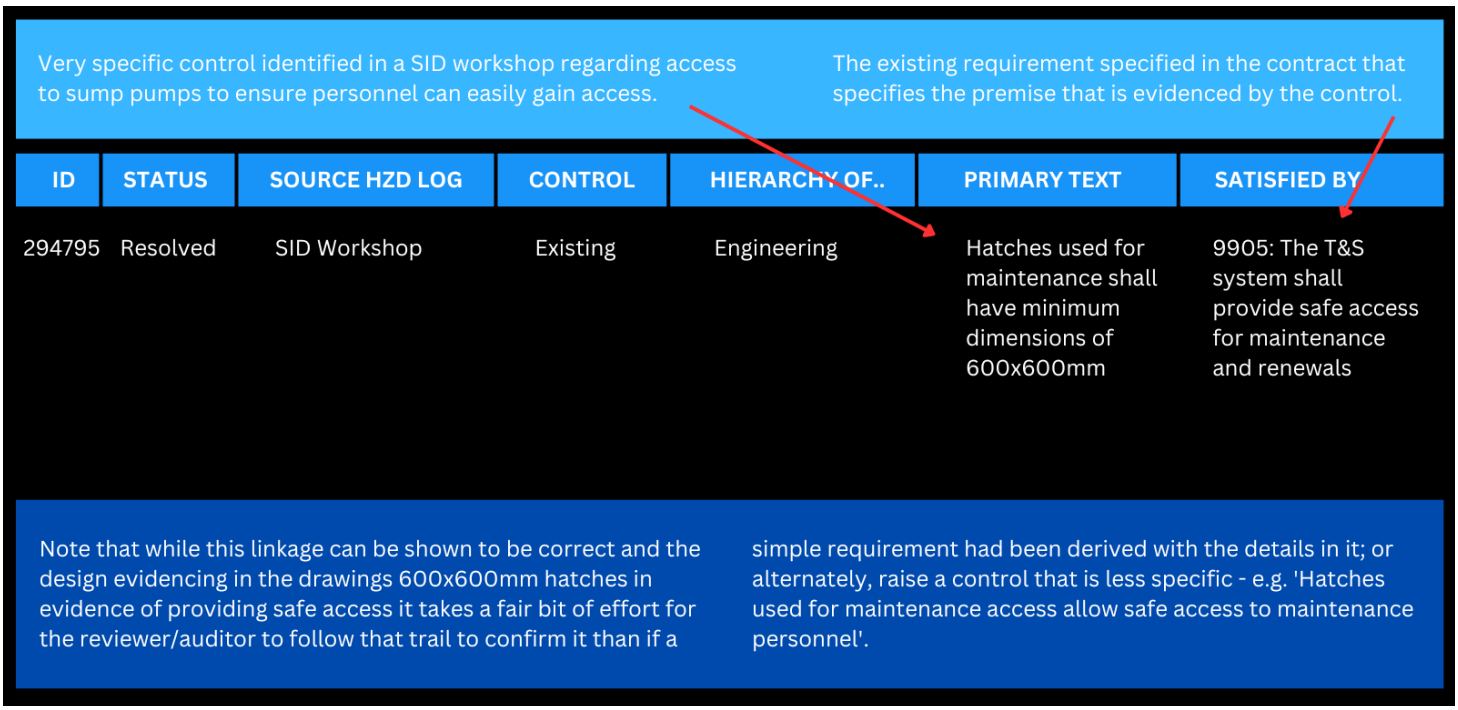
Lack of Competency

A fifth contributing factor the author has observed is a lack of competency across the industry and specifically in the systems safety assurance discipline. One example is a project where a colleague was involved as the Independent Safety Assessor (ISA). The company awarded the work was competent to deliver the design and delivery activities; however, did not understand what was required to deliver a system safety plan, hazard analysis, and safety argument. For example, a project risk register was put forward as the system hazard log.

As such the assurance activities which should have taken a competent practitioner a week or two to complete were drawn out over months with multiple reviews by the ISA. Naturally, this resulted in delays to the project until the client stepped in and provided the assurance support needed for the ISA to sign off that an appropriate level of assurance had been performed. The use of competent assurance personnel will reduce the time and effort spent to achieve the correct output. As shown in the well-known diagram captured in Figure 6 cost and effort increase significantly as time marches on, especially days before commissioning and thus as noted above it is essential to plan early and, in this case, do the work in a timely manner at the right phase of the project.

A second example is in the final assurance checks that another colleague regularly performs for clients prior to a commissioning. These reviews regularly find issues with the evidence for the competency claimed in the submitted paperwork. Whether that is due to expired competency records, claiming a person is a higher level than what the competency systems say, an improperly developed competency management system, lack of a "competent" competency assessor, or simply

Fig. 5: Example control to requirement linkage



competencies missing from the competency systems. This does not necessarily mean that the designers/testers are in fact "not yet competent"; however, when identified at the final assurance gate it causes a lot of last-minute stresses and adds significant complications to obtaining approval for a commissioning to proceed.

It should have been identified and managed at a much earlier point in the delivery. It is in these time-compressed and artificially urgent situations that mistakes are more easily made and the risk of a safety incident increases as shown in Figure 6.

A safe system is delivered by competent personnel and backed by a safety argument developed by competent personnel.

A Solution

So far, the paper has outlined a number of contributing factors the author has observed that commonly lead to the delivery of safety arguments which fail to provide a convincing/compelling argument that the system being delivered is safe. This section presents several principles and supporting examples that the author strives to implement to deliver convincing safety arguments.

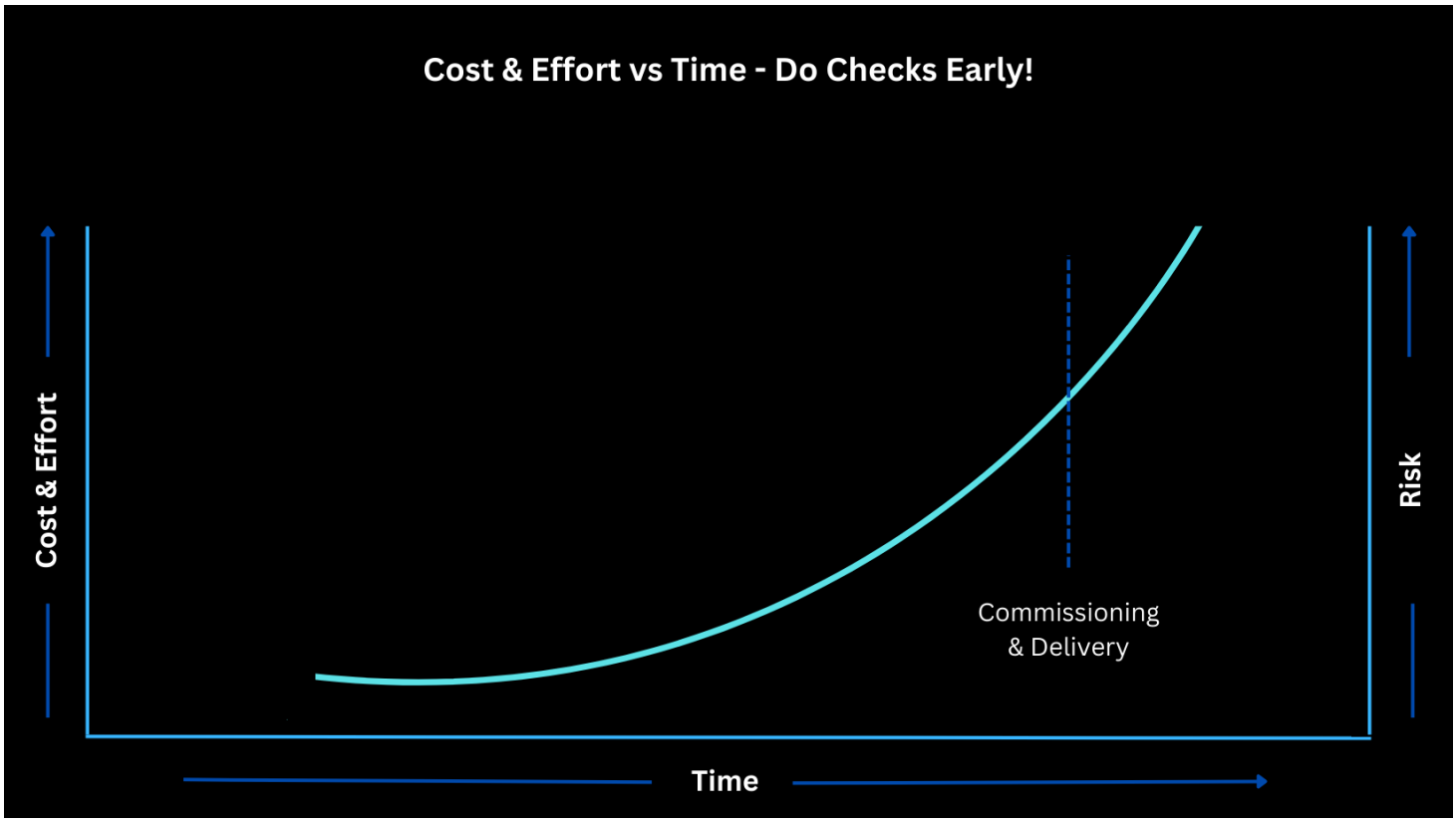
Demonstrate Competency

The first principle is to ensure the project has appropriately competent staff to deliver the job and to capture that evidence in a way that is auditable and can be relied upon as part of your system safety argument. The responsibility for this lies both with the client and with the delivery partners.

As a client, it is very important when specifying a job to make it clear what level of competencies are required and to then follow up and sight the evidence that the delivery partners staff are competent. For the delivery partners it is critical to have competent staff in order ensure a safe and high-quality delivery of the product. Incompetent staff have a detrimental impact to both the client and the delivery partner often causing delays and lower quality outcomes.

Note that basic training in engineering safety management (e.g. iESM [6]) will go a long way to increasing the general competency of an organisation and assessments using frameworks such as that established by the IET [9] will help an organisation establish their own competency frameworks.

Fig. 6: Cost & Effort vs Time



Specifically in the system safety assurance space, it is critical to have a competent and skilled system safety assurance lead who can guide the project from planning through delivery to the final safety case. A pragmatic, competent systems safety assurance lead will be able tailor a systems safety assurance programme to the specific needs of the project, reducing the overall amount of paperwork and time required to deliver the job. As noted above the author regularly sees a lack of competency contributing to re-work and delays.

It is the author's experience that when we can get skilled, competent systems safety assurance engineers (and ideally supported by competent systems engineers) on to a project early we can establish a reasonable, defensible system safety assurance programme that culminates in the delivery of a safe system with a convincing safety argument.

Have a Clear Goal

A second principle is to be clear in what you are going to assure. What are we trying to claim at the end of the system safety assurance programme? For example: the methods, techniques, and focus of analyses will be very different if the goal is to demonstrate high availability of a system rather than the safety of the system.

The argument even changes if we are arguing that a system is safe to construct (i.e. occupational health and safety) compared to presenting an argument that a system is safe to operate and maintain (functional safety).

Both will use similar techniques and structures and even in some cases similar evidence items, but both could have two vastly different conclusions. The more specific the goal the clearer and more tangible the evidence and arguments can become.

For example, an argument for the safety of passengers detraining in a tunnel onto an egress walkway is much easier to make than the argument that the whole railway including a new tunnel and underground stations are safe to operate and maintain. One can be done quite simply through a relatively simple risk assessment while the other will take significantly more work.

The broader the goal the greater the need for a structured argument to collect the various sub-claims/goals, arguments/strategies, and evidence/

solutions required to demonstrate the achievement of the overarching goal.

Using established structures, such as that laid out in recognised standards (e.g. EN50129 [2]), can be quite useful and the use of graphical notation (e.g. Claims, Argument, Evidence (CAE) [4] or Goal Structured Notation (GSN) [3]) can be very helpful when communicating to those unfamiliar with systems safety assurance.

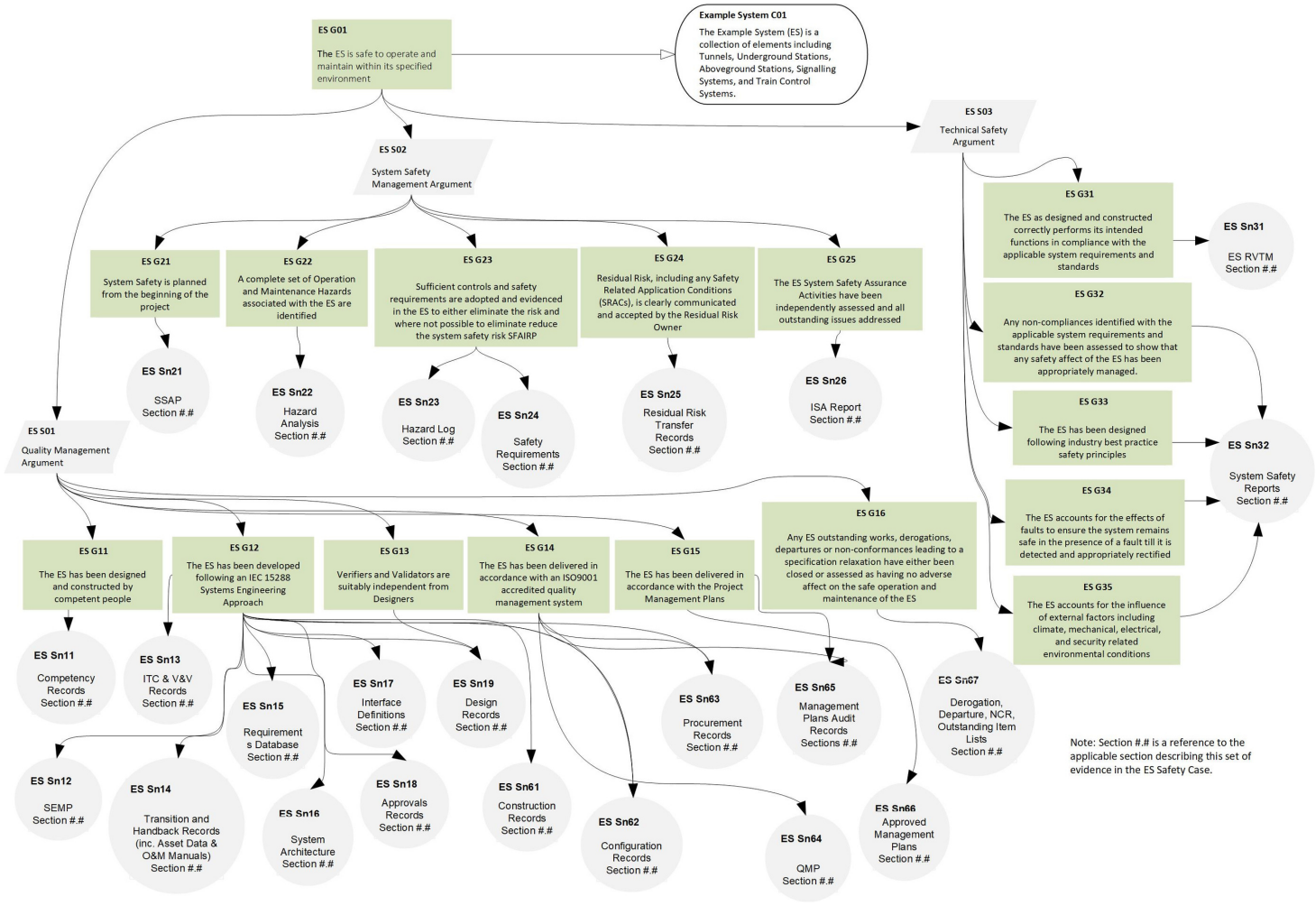
For example, the author has found that a GSN (or CAE) diagram such as shown in Figure 7 that captures the entire argument (based on EN50129 [2]) and which can be printed on an A3 sheet of paper is invaluable to communicate to other members of the project team the importance of their activities in demonstrating that the overall system is safe.

This communication and education piece around the goal and argument to achieve that goal is important in ensuring the resultant argument is a strong well-balanced argument and does not suffer from bloat (see above) or incompleteness. It is important to communicate the argument with all parties involved in demonstrating the argument to ensure they are all on board and understand what is being expected of them.

Keep It Simple

A third principle is to keep your system safety assurance arguments simple. Avoid unnecessary complexity where possible. The more complex they are, the more challenging it is to clearly communicate "why" the system is safe. For example, when delivering changes into a brownfield railway, the more stages required the more complexity is added.

Fig. 7: Example GSN representation of a EN50129 structured safety argument



The more complexity, the more documentation is required and hence the risk increases of confusing baselines, mis-allocating controls, requiring justifications for missing evidence, etc. increases.

If complexity and staging is required, plan your strategy early and, if possible, keep the steps simple and focused on a clear goal. A potential rule of thumb is that if your argument is getting much bigger than what can be easily conveyed on a readable A3 sheet of paper then it is likely straying into unnecessary complexity.

Having said that, there will be times that the nature of the work will necessitate a complex argument, but the principle remains that, if possible, keep the argument simple.

When preparing an argument also be careful to avoid unnecessary walls of text (see above) where a few simple sentences with a table or figure of evidence would suffice.

In keeping the arguments simple it is important to clearly understand the context of the analysis. Be clear on what the system to be assured is, where its boundaries are, what parties are being interfaced with. The argument can also be kept simple by focusing our resources on items that are novel and/or complex, and hence of higher risk.

This does not mean that no analysis is done on less critical systems but that once a system has been determined to not be safety related, or is an existing system, a simple and straight forward argument can be put forward for that system while focusing more effort on the high-risk aspects of the system.

It is also vitally important for clients to not over specify the documentation required to deliver a project. Too often the author sees requests for multiple analyses and reports to be done for various stage gates, which if not handled carefully can lead to unnecessary documentation.

Rather, it is better to specify the minimum requirements and request for a progressive demonstration of assurance evidence as agreed with the project in its systems safety assurance plan.

Write for the Stakeholders

The fourth principle is to tailor the argument for the key stakeholders, which in the Australian Rail industry would be the relevant Rail Transport Operator (RTO) and its Independent Safety Assessor. Know and understand what the key stakeholders expect. Prepare and agree a system safety assurance plan with them and then progressively deliver the activities in the plan.

Fig. 8: Aligning risk matrices

Likelihood Ranking		Consequence Rating				
		Insignificant	Minor	Moderate	Major	Severe
		C1	C2	C3	C4	C5
Certain	L5	M5	M10	H15	E20	E25
Likely	L4	L4	M8	M12	H16	E20
Possible	L3	L3	M6	M9	H12	H15
Unlikely	L2	L2	L4	M6	M8	H10
Rare	L1	L1	L2	L3	M4	H5

≠

Note that while alignment is possible, by comparing definitions, it is rarely perfect and can create significant challenges when getting the final RTO to accept the risk. If translation cannot be agreed it may require reassessment of risk.

Likelihood Ranking		Consequence Rating						
		Very Low	Low	Moderate	High	Very High	Critical	Catastrophic
		C1	C2	C3	C4	C5	C6	C7
Certain	L7	M7	H14	H21	H28	E35	E42	E49
Highly Likely	L6	M6	M12	H18	H24	H30	E36	E42
Likely	L5	M5	M10	M15	H20	H25	H30	E35
Possible	L4	L4	M8	M12	M16	H20	H24	E28
Unlikely	L3	L3	L6	M9	M12	M15	H18	H21
Improbable	L2	VL2	L4	L6	M8	M10	M12	H14
Rare	L1	VL1	VL2	L3	L4	M5	M6	M7

The author finds it invaluable to progressively prepare the safety assurance argument (e.g. safety case) enabling the agreement of the structure and content early in the project and then progressively adding meat to the bones as the project progresses, providing multiple drops in order to address fundamental issues early before the amount of effort required to make changes becomes insurmountable.

An easy example of this is to use the risk matrix of the RTO responsible for operating the railway asset you are delivering. This greatly simplifies the residual risk transfer process required at the end of the project and greatly aids in communication/understanding of the risks (see Figure 8).

It is important to note that the documentation needed to assure one project, while potentially similar, will almost never be appropriate for the next project – even if the project is for the same stakeholders. Therefore, use previous projects for inspiration and to speed the work, but do not fall into the trap of simply copying and pasting as the inevitable errors will only cause confusion and degrade the quality of the argument.

Appropriate Tooling

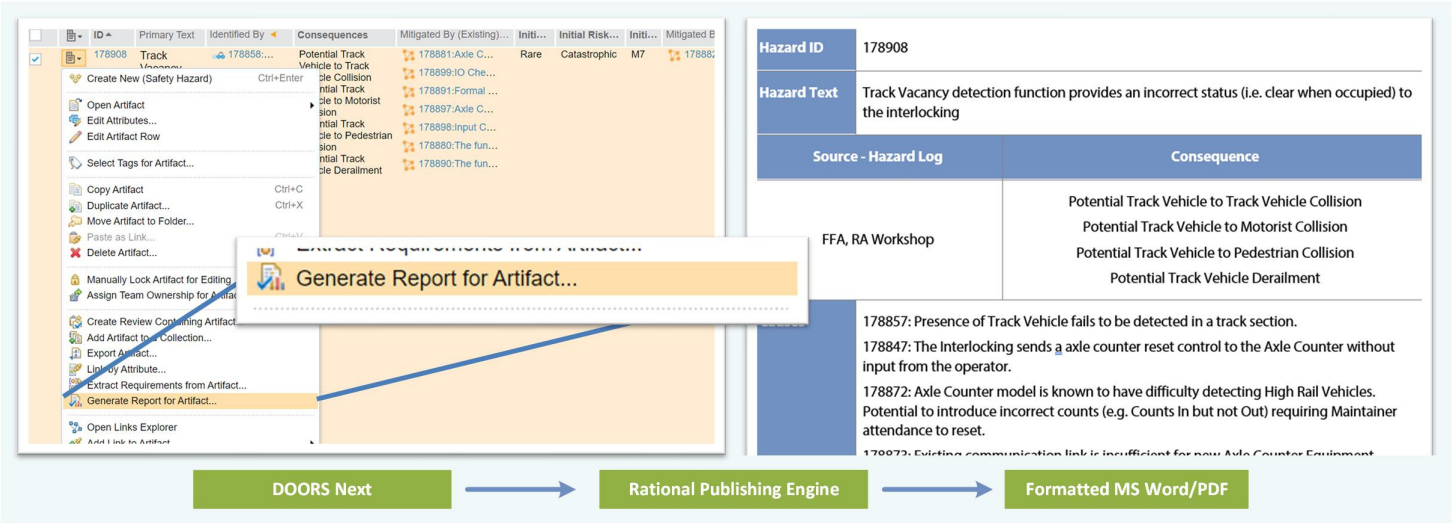
The fifth principle is to use appropriate tools to help you improve the quality and speed the delivery of your overall argument.

For example, in a simple project with 10 to 20 hazards and 100 to 200 causes and controls, a Microsoft Excel sheet may be sufficient to manage the causes/controls/hazards. However, as soon as the project has more than that, or begins to have multiple levels of hazards/causes, then a simple Excel sheet becomes untenable and solutions such as IBM DOORS Next are needed (especially when also being used to manage the project's technical requirements).

When using DOORS Next it is well worth investing in Rationale Publishing Engine to allow you to quickly and easily generate the Hazard Log in a readable and understandable manner. Appropriate tooling will facilitate an "integrated" argument (e.g. System Architecture to Safety Analysis to Hazard Log to System Requirements to Verification and Validation Evidence).

Figure 9 shows an example of the kind of report that can be generated using these tools.

Fig. 9: RPE report example



Another example is the use of appropriate RAMS analysis tools for generating Reliability Block Diagrams, Fault Trees, and Failure Modes and Effects Analysis if such are needed as part of the assurance argument. Using recognised and established tools will save considerable time avoiding arguing over the correctness of calculations and when competently used adds significantly to the pedigree of the argument.

In the context of appropriate tooling, it is important to not re-create the wheel. Use the systems and tools already in place and running on the project for key elements in your argument. For example, if the project is already running to an ISO9001 [7] accredited quality management system or delivering the project to comply with IEC 15288 [8], leverage that for the quality management piece of the safety assurance argument. There is no need to re-create or double up effort.

However, it is important to inform the parties performing the various activities that you are relying on their outputs in the safety argument and planning/agreeing up front what the outcomes are going to look like so you can shape the argument around those.

Conclusion

Writing a good safety argument is no easy task and sadly there is no silver bullet to make it work perfectly. Needless to say, the answer is not requiring more paperwork but in planning early and delivering high quality work by competent people.

It involves engaging stakeholders early, setting expectations, and establishing clear goal posts. A good safety argument is structured, well balanced, and well proportioned.

It is simple, complete, structured, and to the point. It focuses on areas of high risk and or novelty while still clearly articulating the argument for the business as usual. It has accessible and auditable evidence substantiating each claim. It is supported by a well-structured hazard log and reputable tools. It relies on a system delivered to established standards and by competent personnel.

An argument that follows these key principles will be able to deliver a safety argument for a system that is clear, concise, and cost effective. An argument that is tailored for the system in question and when read leaves the reader with a clear understanding of "why" the system is safe.

Andrew Gabler | Principal Consultant

References

1. Rail Safety National Law (RSNL) Section 46, 47, and 53.
2. Railway applications - Communication, signalling and processing systems - Safety related electronic systems for signalling, EN50129, European Committee for Electrotechnical Standardization (CENELEC), 2018
3. Goal Structuring Notation (GSN), https://www.adelard.com/asce/gsn/
4. Claims, Arguments, and Evidence (CAE), https://www.adelard.com/asce/cae/
5. Railway applications - The Specification and Demonstration of Reliability, Availability, Maintainability and Safety (RAMS) - Part 1: Generic RAMS process, EN50126-1, European Committee for Electrotechnical Standardization (CENELEC), 2017
6. International Engineering Management (iESM), https://www.intesm.org/
7. Quality Management Systems, ISO9001, International Organization for Standarization, 2015
8. Systems and software engineering – System lifecycle processes, ISO/IEC/IEEE 15288, International Electrotechnical Commission, 2015
9. Code of Practice: Competence for Safety Related Systems Practitioners, The Institution of Engineering and Technology, https://shop.theiet.org/code-of-practice-competence-for-safety-related-systems-practitioners

Acmena

CONTACT US

+61 (0) 478 814 324
enquiries@acmena.com.au
www.acmena.com.au

Acmena Group Pty Ltd
PO BOX 220
Ashgrove West
Brisbane, QLD 4060
ABN: 37 158 514955
ACN: 158 514 955